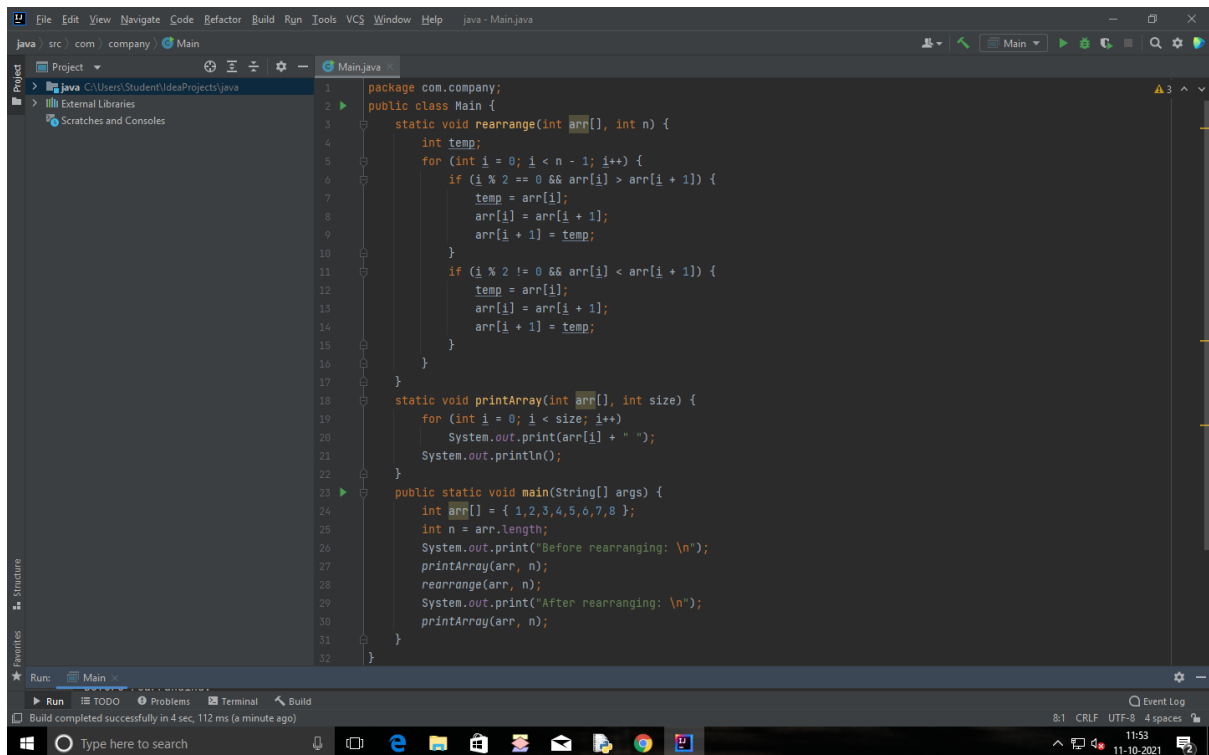1.)Design an algorithm that rearranges the elements of an array so that all those originally stored at odd suffixes are placed before those at even suffixes

Input : 8

1 2 3 4 5 6 7 8

Output :1 3 5 7 2 4 6 8


```java
package com.company;
public class Main {
    static void rearrange(int arr[], int n) {
        int temp;
        for (int i = 0; i < n - 1; i++) {
            if (i % 2 == 0 && arr[i] > arr[i + 1]) {
                temp = arr[i];
                arr[i] = arr[i + 1];
                arr[i + 1] = temp;
            }
            if (i % 2 != 0 && arr[i] < arr[i + 1]) {
                temp = arr[i];
                arr[i] = arr[i + 1];
                arr[i + 1] = temp;
            }
        }
    }
    static void printArray(int arr[], int size) {
        for (int i = 0; i < size; i++)
            System.out.print(arr[i] + " ");
        System.out.println();
    }
    public static void main(String[] args) {
        int arr[] = { 1,2,3,4,5,6,7,8 };
        int n = arr.length;
        System.out.print("Before rearranging: \n");
        printArray(arr, n);
        rearrange(arr, n);
        System.out.print("After rearranging: \n");
        printArray(arr, n);
    }
}
```

```java
            }
        }
    }
    static void printArray(int arr[], int size) {
        for (int i = 0; i < size; i++)
            System.out.print(arr[i] + " ");
        System.out.println();
    }
    public static void main(String[] args) {
        int arr[] = { 1,2,3,4,5,6,7,8 };
        int n = arr.length;
        System.out.print("Before rearranging: \n");
        printArray(arr, n);
        rearrange(arr, n);
        System.out.print("After rearranging: \n");
        printArray(arr, n);
    }
}
```

```
"C:\Program Files\Java\jdk1.8.0_131\bin\java.exe" ...
Before rearranging:
1 2 3 4 5 6 7 8
After rearranging:
1 3 2 5 4 7 6 8

Process finished with exit code 0
```